

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

JPO72 U.S. PTO  
10/051276  
01/22/02

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2001年 1月22日

出 願 番 号

Application Number:

特願2001-013565

出 願 人

Applicant(s):

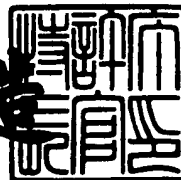
株式会社東芝

CERTIFIED COPY OF  
PRIORITY DOCUMENT

2001年 9月13日

特許庁長官  
Commissioner,  
Japan Patent Office

及川耕造



出証番号 出証特2001-308463

【書類名】 特許願

【整理番号】 A000007749

【提出日】 平成13年 1月22日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 15/00

【発明の名称】 べき乗剰余計算装置、べき乗剰余計算方法及び記録媒体

【請求項の数】 16

【発明者】

    【住所又は居所】 神奈川県川崎市幸区小向東芝町 1 番地 株式会社東芝研  
究開発センター内

    【氏名】 新保 淳

【発明者】

    【住所又は居所】 神奈川県川崎市幸区小向東芝町 1 番地 株式会社東芝研  
究開発センター内

    【氏名】 池田 華恵

【特許出願人】

    【識別番号】 000003078

    【氏名又は名称】 株式会社 東芝

【代理人】

    【識別番号】 100058479

    【弁理士】

    【氏名又は名称】 鈴江 武彦

    【電話番号】 03-3502-3181

【選任した代理人】

    【識別番号】 100084618

    【弁理士】

    【氏名又は名称】 村松 貞男

【選任した代理人】

    【識別番号】 100068814

【弁理士】

【氏名又は名称】 坪井 淳

【選任した代理人】

【識別番号】 100092196

【弁理士】

【氏名又は名称】 橋本 良郎

【選任した代理人】

【識別番号】 100091351

【弁理士】

【氏名又は名称】 河野 哲

【選任した代理人】

【識別番号】 100088683

【弁理士】

【氏名又は名称】 中村 誠

【選任した代理人】

【識別番号】 100070437

【弁理士】

【氏名又は名称】 河井 将次

【手数料の表示】

【予納台帳番号】 011567

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 べき乗剰余計算装置、べき乗剰余計算方法及び記録媒体

【特許請求の範囲】

【請求項1】

対象データC及び独立パラメータp、q、dについて、複数の整数の組からなる第1の基底及び第2の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第1の基底の全整数の積 $A > p$ 、q、且つ、第2の基底の全整数の積 $B > p$ 、q、且つ、 $A * B > C$ とする）、計算結果 $m = C^d \bmod p * q$ を求めるためのべき乗剰余計算装置であって、

前記データCのpによる剰余値 $C_p$ の剰余系表現及び前記パラメータdの $(p-1)$ による剰余値 $d_p$ に基づき、 $(C_p^{d_p}) * B \bmod p$ 又はこれにpが加えられた値の剰余系表現を求めるための第1の処理手段と、

前記データCのqによる剰余値 $C_q$ の剰余系表現及び前記パラメータdの $(q-1)$ による剰余値 $d_q$ に基づき、 $(C_q^{d_q}) * B \bmod q$ 又はこれにqが加えられた値の剰余系表現を求めるための第2の処理手段と、

前記第1及び第2の処理手段により求められた両剰余系表現に基づき、法 $p * q$ の元で $C^d$ と合同である整数 $m'$ の剰余系表現を求めるための第3の処理手段と、

前記第3の処理手段により求められた前記剰余系表現を2進数表現に変換して得られた前記整数 $m'$ の値に基づき、前記計算結果mを求めるための第4の処理手段とを備えたことを特徴とするべき乗剰余計算装置。

【請求項2】

前記第1の処理手段は、前記剰余値 $C_p = C \bmod p$ の剰余系表現と $B^2 \bmod p$ の剰余系表現とのRNSモンゴメリ乗算を行い、これにより得られた剰余系表現について指数部を前記剰余値 $d_p = d \bmod (p-1)$ とするRNSモンゴメリべき乗を行うことによって、 $(C_p^{d_p}) * B \bmod p$ 又はこれにpが加えられた値の剰余系表現を求め、

前記第2の処理手段は、前記剰余値 $C_q = C \bmod q$ の剰余系表現と $B^2 \bmod q$ の剰余系表現とのRNSモンゴメリ乗算を行い、これにより得ら

れた剰余系表現について指数部を前記剰余値  $d_q = d \bmod (q-1)$  とするRNSモンゴメリべき乗を行うことによって、 $(C_q^{d_q}) * B \bmod q$  又はこれに  $q$  が加えられた値の剰余系表現を求めることを特徴とする請求項1に記載のべき乗剰余計算装置。

### 【請求項3】

前記パラメータ  $p$ 、 $q$  及び  $d$  をもとに、前記剰余値  $d_p = d \bmod (p-1)$  及び前記剰余値  $d_q = d \bmod (q-1)$  を求めるための手段を更に備えたことを特徴とする請求項2に記載のべき乗剰余計算装置。

### 【請求項4】

前記第3の処理手段は、前記第1の処理手段により得られた前記剰余系表現と前記パラメータ  $q$  の法  $p$  における逆元  $q_{inv} = q^{-1} \bmod p$  の剰余系表現とのRNSモンゴメリ乗算を行い、これによって得られた剰余系表現と前記パラメータ  $q$  の剰余系表現とのRNS乗算を行うとともに、前記第2の処理手段により得られた前記剰余系表現と前記パラメータ  $p$  の法  $q$  における逆元  $p_{inv} = p^{-1} \bmod q$  の剰余系表現とのRNSモンゴメリ乗算を行い、これによって得られた剰余系表現と前記パラメータ  $p$  の剰余系表現とのRNS乗算を行い、これらにより得られた両RNS乗算の結果のRNS加算を行うことによって、前記法  $p * q$  において  $C^d$  と合同である整数  $m'$  の剰余系表現を求めることを特徴とする請求項1に記載のべき乗剰余計算装置。

### 【請求項5】

前記パラメータ  $p$  及び前記パラメータ  $q$  並びに前記逆元  $p_{inv}$  及び前記逆元  $q_{inv}$  をそれぞれ2進数表現から剰余系表現に変換するための手段を更に備えたことを特徴とする請求項4に記載のべき乗剰余計算装置。

### 【請求項6】

前記パラメータ  $p$  及び  $q$  をもとに、前記逆元  $p_{inv} = p^{-1} \bmod q$  及び前記パラメータ  $q$  の法  $p$  における逆元  $q_{inv} = q^{-1} \bmod p$  を求めるための手段を更に備えたことを特徴とする請求項5に記載のべき乗剰余計算装置。

### 【請求項7】

前記データC並びに前記パラメータp及びqをもとに、前記剰余値 $C_p = C \bmod p$ 及び前記剰余値 $C_q = C \bmod q$ を求めるための手段を更に備えたことを特徴とする請求項1に記載のべき乗剰余計算装置。

【請求項8】

前記パラメータp、q、dのみに依存する剰余系表現のデータを予め記憶しておくための記憶手段を更に備えたことを特徴とする請求項1、2または4に記載のべき乗剰余計算装置。

【請求項9】

前記パラメータを識別する識別情報iと、該識別情報iに対応するパラメータ $p_i$ 、 $q_i$ 、 $d_i$ のみに依存する剰余系表現のデータとを対応付けて予め記憶しておくための記憶手段を更に備えたことを特徴とする請求項1、2または4に記載のべき乗剰余計算装置。

【請求項10】

前記第1の処理手段及び第2の処理手段はそれらの処理の少なくとも一部を同時に実行することを特徴とする請求項1に記載のべき乗剰余計算装置。

【請求項11】

前記第1の処理手段及び第2の処理手段は前記基底の要素ごとに行う演算につき該要素の数に相当する演算の全部または一部を同時に実行することを特徴とする請求項1に記載のべき乗剰余計算装置。

【請求項12】

前記第4の処理手段は、

前記第3の処理手段により求められた前記整数 $m'$ の剰余系表現を2進数表現に変換するための手段と、

この手段により得られた $p * q$ 未満の前記整数 $m'$ の値又は $p * q$ 以上の前記整数 $m'$ から所定回数 $p * q$ を減じることによって得た $p * q$ 未満の値を、 $m = C^d \bmod p * q$ とするための手段とを含むことを特徴とする請求項1に記載のべき乗剰余計算装置。

【請求項13】

前記第1の基底の要素数と前記第2の基底の要素数とを同一にしたことを特徴

とする請求項1に記載のべき乗剰余計算装置。

【請求項14】

対象データC及び独立パラメータp、q、dについて、複数の整数の組からなる第1の基底及び第2の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第1の基底の全整数の積 $A > p$ 、q、且つ、第2の基底の全整数の積 $B > p$ 、q、且つ、 $A * B > C$ とする）、 $m = C^d \bmod p * q$ を求めるためのべき乗剰余計算方法であって、

前記データCのpによる剰余値 $C_p$ の剰余系表現及び前記パラメータdの $(p-1)$ による剰余値 $d_p$ に基づき、 $(C_p^{d_p}) * B \bmod p$ 又はこれにpが加えられた値の剰余系表現を求めるとともに、前記データCのqによる剰余値 $C_q$ の剰余系表現及び前記パラメータdの $(q-1)$ による剰余値 $d_q$ に基づき、 $(C_q^{d_q}) * B \bmod q$ 又はこれにqが加えられた値の剰余系表現を求め、

求められた両剰余系表現に基づき、法 $p * q$ の元で $C^d$ と合同である整数 $m'$ の剰余系表現を求め、

求められた前記剰余系表現を2進数表現に変換して得られた前記整数 $m'$ の値に基づき、前記計算結果mを求めることを特徴とするべき乗剰余計算方法。

【請求項15】

対象データC及び独立パラメータp、q、dについて、複数の整数の組からなる第1の基底及び第2の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第1の基底の全整数の積 $A > p$ 、q、且つ、第2の基底の全整数の積 $B > p$ 、q、且つ、 $A * B > C$ とする）、 $m = C^d \bmod p * q$ を求めるためのべき乗剰余計算装置としてコンピュータを機能させるためのプログラムを記録したコンピュータ読取り可能な記録媒体であって、

前記データCのpによる剰余値 $C_p$ の剰余系表現及び前記パラメータdの $(p-1)$ による剰余値 $d_p$ に基づき、 $(C_p^{d_p}) * B \bmod p$ 又はこれにpが加えられた値の剰余系表現を求めるための第1の処理機能と、

前記データCのqによる剰余値 $C_q$ の剰余系表現及び前記パラメータdの $(q-1)$ による剰余値 $d_q$ に基づき、 $(C_q^{d_q}) * B \bmod q$ 又はこれに

q が加えられた値の剰余系表現を求めるための第2の処理機能と、

前記第1及び第2の処理機能により求められた両剰余系表現に基づき、法  $p * q$  の元で  $C^d$  と合同である整数  $m'$  の剰余系表現を求めるための第3の処理機能と、

前記第3の処理機能により求められた前記剰余系表現を2進数表現に変換して得られた前記整数  $m'$  の値に基づき、前記計算結果  $m$  を求めるための第4の処理機能とをコンピュータに実現させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【請求項16】

対象データ  $C$  及び独立パラメータ  $p$ 、 $q$ 、 $d$  について、複数の整数の組からなる第1の基底及び第2の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第1の基底の全整数の積  $A > p$ 、 $q$ 、且つ、第2の基底の全整数の積  $B > p$ 、 $q$ 、且つ、 $A * B > C$  とする）、 $m = C^d \bmod p * q$  を求めるためのべき乗剰余計算装置としてコンピュータを機能させるためのプログラム、

前記データ  $C$  の  $p$  による剰余値  $C_p$  の剰余系表現及び前記パラメータ  $d$  の  $(p-1)$  による剰余値  $d_p$  に基づき、 $(C_p^{d_p}) * B \bmod p$  又はこれに  $p$  が加えられた値の剰余系表現を求めるための第1の処理機能と、

前記データ  $C$  の  $q$  による剰余値  $C_q$  の剰余系表現及び前記パラメータ  $d$  の  $(q-1)$  による剰余値  $d_q$  に基づき、 $(C_q^{d_q}) * B \bmod q$  又はこれに  $q$  が加えられた値の剰余系表現を求めるための第2の処理機能と、

前記第1及び第2の処理機能により求められた両剰余系表現に基づき、法  $p * q$  の元で  $C^d$  と合同である整数  $m'$  の剰余系表現を求めるための第3の処理機能と、

前記第3の処理機能により求められた前記剰余系表現を2進数表現に変換して得られた前記整数  $m'$  の値に基づき、前記計算結果  $m$  を求めるための第4の処理機能とをコンピュータに実現させるためのプログラム。

【発明の詳細な説明】

【0001】



## 【発明の属する技術分野】

本発明は、対象データ  $C$  及び独立パラメータ  $p$ 、 $q$ 、 $d$  について  $m = C^d \bmod p * q$  を求めるためのべき乗剰余計算装置及びべき乗剰余計算方法に関する。

## 【0002】

## 【従来の技術】

整数演算（加減乗算）の並列処理が可能となる剰余系（Residue Number System: RNS）表現をベースに、公開鍵暗号のアルゴリズム（べき乗剰余演算）を実現するための基本要素となる剰余乗算を、モンゴメリ乗算と融合させて実現するアルゴリズムおよびそのハードウェア構成が提案されている。これをRNSモンゴメリ乗算と呼ぶことにする。

## 【0003】

ここで、RNS表現（剰余系表現）について説明する。RSA暗号などの公開鍵暗号の多くでは多倍長整数を利用して変換を行うが、多倍長整数の表現に通常利用されるのは基数を2とした記数法（Radix representation）、いわゆる2進数表現である。これとは別の方法に、複数の法  $a_1, a_2, \dots, a_n$  を用意し、整数  $x$  をそれらの法による剰余値  $x_1, x_2, \dots, x_n$  の組で表現する方法がある。すなわち、以下の式による。

$$x_1 = x \bmod a_1, x_2 = x \bmod a_2, \dots, x_n = x \bmod a_n$$

この表現法はRNS表現と呼ばれる。

## 【0004】

以下では、RNS表現で用いる法の集合を、基底（base）と呼ぶものとする。また、基底の要素数  $n$  を基底サイズと呼ぶものとする。基底サイズ  $n$  の基底  $a$  は、次のように表される。

$$a = \{a_1, a_2, \dots, a_n\}$$

RNS表現では、各基底の要素は、通常、互いに素な正整数を用い、中国剰余定理により“基底の要素の積”未満の正整数はRNS表現により一意に表現できることが保証される。すなわち、

基底  $a = \{a_1, a_2, \dots, a_n\}$

基底  $a$  の要素の積  $A = a_1 * a_2 * \dots * a_n$

としたときに、基底  $a$  を用いた RNS 表現により  $A$  未満の正整数を表現できる。

【0005】

以下では、基底  $a$  を用いて RNS 表現された整数  $x$  を、 $\langle x \rangle_a$  で表すものとする（基底を省略して  $\langle x \rangle$  で表すこともある）。すなわち、

$$\begin{aligned}\langle x \rangle_a &= (x_1, x_2, \dots, x_n) \\ &= (x \bmod a_1, x \bmod a_2, \dots, x \bmod a_n)\end{aligned}$$

である。

【0006】

なお、以下の演算で、2種類の基底を用いる場合に、基底  $a = \{a_1, a_2, \dots, a_{n1}\}$  と基底  $b = \{b_1, b_2, \dots, b_{n2}\}$  について、 $a \cup b$  は、 $\{a_1, a_2, \dots, a_{n1}\}$  と  $\{b_1, b_2, \dots, b_{n2}\}$  の結合を表し、 $\langle x \rangle_{a \cup b}$  は、基底  $a \cup b$  による  $x$  の RNS 表現を表す（すなわち、 $\langle x \rangle_{a \cup b}$  は、 $\langle x \rangle_a = (x \bmod a_1, x \bmod a_2, \dots, x \bmod a_{n1})$  と  $\langle x \rangle_b = (x \bmod b_1, x \bmod b_2, \dots, x \bmod b_{n2})$  の結合を表す）。また、この場合に、2種類の基底  $n1 = n2 = n$  として説明する。

【0007】

RNS 表現の利点は、基底の全要素の積  $A$  を法とした加算、減算、乗算が簡単に計算できることである。すなわち、以下のように各要素をそれぞれの法によって独立に加算、減算、乗算した結果が、所望の結果となる。

$$\langle x \rangle_a + \langle y \rangle_a \pmod{A} = (x_1 + y_1 \pmod{a_1}, x_2 + y_2 \pmod{a_2}, \dots, x_n + y_n \pmod{a_n})$$

$$\langle x \rangle_a - \langle y \rangle_a \pmod{A} = (x_1 - y_1 \pmod{a_1}, x_2 - y_2 \pmod{a_2}, \dots, x_n - y_n \pmod{a_n})$$

$$\langle x \rangle_a * \langle y \rangle_a \pmod{A} = (x_1 * y_1 \pmod{a_1}, x_2 * y_2 \pmod{a_2}, \dots, x_n * y_n \pmod{a_n})$$

なお、上記の演算を、それぞれ、RNS 加算、RNS 減算、RNS 乗算と呼ぶものとする。

## 【0008】

従って、 $n$ 個の演算はすべて並列に処理可能であるため、 $n$ 個の演算ユニットを用意すればすべてを並列に処理して高速な処理が行えるし、用意する演算ユニットが $n$ 個に満たなくても1個から $n$ 個まで増やす程それに比例して演算速度を向上できる。

## 【0009】

次に、RNSモンゴメリ乗算およびRNSモンゴメリべき乗について説明する。

## 【0010】

RNSモンゴメリ乗算は、法 $p$ での剰余付き乗算( $\langle x \rangle * \langle y \rangle \pmod{N}$ )に関し、モンゴメリ乗算と呼ばれる手法をRNS表現での演算に応用する手法であり、概ね以下の手順で計算される。

## 【0011】

[RNSモンゴメリ乗算:  $MM(\langle x \rangle_{aUb}, \langle y \rangle_{aUb}, N, aUb)$ ]

入力:  $\langle x \rangle_{aUb}, \langle y \rangle_{aUb}, N$

ただし、 $x, y < 2N$

基底:  $a, b$

ただし、 $x, y, N < A, B$

出力:  $\langle w \rangle_{aUb}$

ただし、 $w = x * y * B^{-1} \pmod{N}$

<処理内容>

Step-M-0:  $\langle (-N^{-1}) \rangle_b$ を計算する。

Step-M-1:  $\langle s \rangle_a = \langle x \rangle_a * \langle y \rangle_a$ を計算する。

Step-M-2:  $\langle s \rangle_b = \langle x \rangle_b * \langle y \rangle_b$ を計算する。

Step-M-3:  $\langle t \rangle_b = \langle s \rangle_b * \langle (-N^{-1}) \rangle_b$ を計算する。

Step-M-4:  $\langle t \rangle_b$ を $\langle t \rangle_a$ へ基底変換する。

Step-M-5:  $\langle u \rangle_a = \langle t \rangle_a * \langle N \rangle_a$ を計算する。

Step-M-6 :  $\langle v \rangle_a = \langle s \rangle_a + \langle u \rangle_a$  を計算する。

Step-M-7 :  $\langle w \rangle_a = \langle v \rangle_a * \langle B^{-1} \rangle_a$  を計算する。

Step-M-8 :  $\langle w \rangle_a$  を  $\langle w \rangle_b$  へ基底変換する。

【0012】

なお、上記の手順のうちStep-M-4 やStep-M-8 の基底変換は、ある基底によるRNS表現に対応するある整数（例えば、基底bによるRNS表現 $\langle t \rangle_b$ に対応する整数t）の、他の基底によるRNS表現（例えば、基底aによるRNS表現 $\langle t \rangle_a$ ）を求める処理である。

【0013】

RNSモンゴメリ乗算器も並列に処理を行う演算ユニットを増加させることにより、高速処理が可能となる。

【0014】

また、RNSモンゴメリ乗算を繰り返し行うことで（RNSモンゴメリ乗算器を繰り返し利用することで）べき乗計算を行うことによりRSA暗号の暗号処理を構成する方法が提案されている。このべき乗計算法をRNSモンゴメリべき乗と呼ぶものとする。RNSモンゴメリべき乗は、概ね以下の手順で計算される。

【0015】

[RNSモンゴメリべき乗 :  $MEXP(\langle x \rangle_a U b, d, N, a U b)$ ]

入力 :  $\langle x \rangle_a U b$ , 指数  $d = (d_k, d_{k-1}, \dots, d_1)$   
 $)_2$ 、法N

ただし、 $x < 2N$

基底 : a, b

ただし、 $x, N < A, B$

出力 :  $\langle y \rangle_a U b$

ただし、 $y = x^d * B^{-(d-1)} \bmod N$

<処理内容>

Step-E-1 :  $i = k$  とする。 $\langle y \rangle_a U b = \langle B \rangle_a U b$  とする。

Step-E-2 :  $\langle y \rangle_a U b = MM(\langle y \rangle_a U b, \langle y \rangle_a U b, N$

,  $a \cup b$ ) を計算する。

Step-E-3 :  $d\_i = 1$  ならば、 $\langle y \rangle\_a \cup b = MM(\langle y \rangle\_a \cup b, \langle x \rangle\_a \cup b, N, a \cup b)$  を計算する。 $d\_i \neq 1$  ならば、なにもしない (nop)。

Step-E-4 :  $i = i - 1$  とする。

Step-E-5 :  $i = 0$  ならば、終了する。 $i \neq 0$  ならば、Step-E-2 へ戻る。

【0016】

なお、上記の手順のうち、Step-E-2 や Step-E-3 での  $MM( )$  は、先に説明した RNS モンゴメリ乗算を表す。

【0017】

次に、CRT べき乗剰余計算について説明する。

【0018】

RSA 暗号は、公開鍵 ( $N, e$ )、秘密鍵 ( $d, p, q$ ) に対し、 $C = m^e \bmod N$  で平文  $m$  を暗号文  $C$  に暗号変換し、 $m = C^d \bmod N$  で暗号文  $C$  を平文  $m$  に復号変換する。ここで、公開鍵である法  $N$  の秘密素因数  $p, q$  を利用して復号変換をより効率的に実行する計算法である、中国剰余定理 (Chinese Remainder Theorem: CRT) を利用したべき乗計算法が知られている。このようなべき乗計算法を、CRT べき乗剰余計算と呼ぶものとする。

【0019】

<CRT べき乗剰余計算手順>

(Step-C-1)  $d_p = d \bmod (p-1)$

$d_q = d \bmod (q-1)$

(Step-C-2)  $C_p = C \bmod p$

$C_q = C \bmod q$

(Step-C-3)  $m_p = C_p^{d_p} \bmod p$

$m_q = C_q^{d_q} \bmod q$

(Step-C-4)  $m = m_p * (q^{(-1) \bmod p}) * q +$

$m_q * (p^{(-1) \bmod q}) * p \pmod{N}$

なお、上記の手順において、 $d_p$ 、 $d_q$ 、 $(q^{-1} \bmod p)$ 、 $(p^{-1} \bmod q)$ といったパラメータは秘密鍵のみに依存するので、事前に計算し、秘密鍵の一部として記憶しておくことが一般的である。

#### 【0020】

CRTべき乗剰余計算の計算量は、その支配的な部分が(Step-C-3)のべき乗剰余計算2回であり、べき乗剰余計算は法のサイズの3乗に比例することに注意すると、2進表現でのべき乗剰余計算とCRTべき乗剰余計算の計算量は約1/4であることが分かる。なお、(Step-C-3)のべき乗計算を2つの計算回路で同時に実行すれば計算時間は約1/8にできる。

#### 【0021】

しかしながら、RNSモンゴメリ乗算によるRSA暗号装置においてCRTべき乗剰余計算の具体的な手順は示されておらず、RSA復号変換(秘密変換)を同装置でより短時間に計算することはできなかった。

#### 【0022】

##### 【発明が解決しようとする課題】

従来、RNSモンゴメリ乗算によるCRTべき乗剰余計算の具体的な手順が知られていなかったため、RSA復号変換(秘密変換)のような大きな整数のべき乗剰余計算をより高速化することが困難であった。

#### 【0023】

本発明は、上記事情を考慮してなされたもので、べき乗剰余計算をより効率良く実行可能にしたべき乗剰余計算装置及びべき乗剰余計算方法を提供することを目的とする。

#### 【0024】

##### 【課題を解決するための手段】

本発明は、対象データC及び独立パラメータp、q、dについて、複数の整数の組からなる第1の基底及び第2の基底による剰余系表現を利用して(ただし、両基底に含まれる全整数は互いに素、且つ、第1の基底の全整数の積 $A > p$ 、q、且つ、第2の基底の全整数の積 $B > p$ 、q、且つ、 $A * B > C$ とする)、計算結果 $m = C^d \bmod p * q$ を求めるためのべき乗剰余計算装置であって、

前記データCのpによる剰余値 $C_p$ の剰余系表現及び前記パラメータdの $(p-1)$ による剰余値 $d_p$ に基づき、 $(C_p^{d_p}) * B \bmod p$ 又はこれにpが加えられた値の剰余系表現を求めるための第1の処理手段と、前記データCのqによる剰余値 $C_q$ の剰余系表現及び前記パラメータdの $(q-1)$ による剰余値 $d_q$ に基づき、 $(C_q^{d_q}) * B \bmod q$ 又はこれにqが加えられた値の剰余系表現を求めるための第2の処理手段と、前記第1及び第2の処理手段により求められた両剰余系表現に基づき、法 $p * q$ の元で $C^d$ と合同である整数 $m'$ の剰余系表現を求めるための第3の処理手段と、前記第3の処理手段により求められた前記剰余系表現を2進数表現に変換して得られた前記整数 $m'$ の値に基づき、前記計算結果mを求めるための第4の処理手段とを備えたことを特徴とする。

## 【0025】

好ましくは、前記第1の処理手段は、前記剰余値 $C_p = C \bmod p$ の剰余系表現と $B^2 \bmod p$ の剰余系表現とのRNSモンゴメリ乗算を行い、これにより得られた剰余系表現について指数部を前記剰余値 $d_p = d \bmod (p-1)$ とするRNSモンゴメリべき乗を行うことによって、 $(C_p^{d_p}) * B \bmod p$ 又はこれにpが加えられた値の剰余系表現を求め、前記第2の処理手段は、前記剰余値 $C_q = C \bmod q$ の剰余系表現と $B^2 \bmod q$ の剰余系表現とのRNSモンゴメリ乗算を行い、これにより得られた剰余系表現について指数部を前記剰余値 $d_q = d \bmod (q-1)$ とするRNSモンゴメリべき乗を行うことによって、 $(C_q^{d_q}) * B \bmod q$ 又はこれにqが加えられた値の剰余系表現を求めるようにしてもよい。

## 【0026】

好ましくは、前記第3の処理手段は、前記第1の処理手段により得られた前記剰余系表現と前記パラメータqの法pにおける逆元 $q_{inv} = q^{(-1)} \bmod p$ の剰余系表現とのRNSモンゴメリ乗算を行い、これによって得られた剰余系表現と前記パラメータqの剰余系表現とのRNS乗算を行うとともに、前記第2の処理手段により得られた前記剰余系表現と前記パラメータpの法qにおける逆元 $p_{inv} = p^{(-1)} \bmod q$ の剰余系表現とのRNSモンゴメ

り乗算を行い、これによって得られた剰余系表現と前記パラメータ  $p$  の剰余系表現との RNS 乗算を行い、これらにより得られた両 RNS 乗算の結果の RNS 加算を行うことによって、前記法  $p * q$  において  $C^d$  と合同である整数  $m'$  の剰余系表現を求めるようにしてもよい。

【0027】

好ましくは、前記第 1 の処理手段及び第 2 の処理手段はそれらの処理の少なくとも一部を同時に実行するようにしてもよい。

【0028】

好ましくは、前記第 1 の処理手段及び第 2 の処理手段は前記基底の要素ごとに行う演算につき該要素の数に相当する演算の全部または一部を同時に実行するようにしてもよい。

【0029】

好ましくは、前記第 4 の処理手段は、前記第 3 の処理手段により求められた前記整数  $m'$  の剰余系表現を 2 進数表現に変換するための手段と、この手段により得られた  $p * q$  未満の前記整数  $m'$  の値又は  $p * q$  以上の前記整数  $m'$  から所定回数  $p * q$  を減じることによって得た  $p * q$  未満の値を、 $m = C^d \bmod p * q$  とするための手段とを含むようにしてもよい。

【0030】

また、本発明は、対象データ  $C$  及び独立パラメータ  $p$ 、 $q$ 、 $d$  について、複数の整数の組からなる第 1 の基底及び第 2 の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第 1 の基底の全整数の積  $A > p$ 、 $q$ 、且つ、第 2 の基底の全整数の積  $B > p$ 、 $q$ 、且つ、 $A * B > C$  とする）、 $m = C^d \bmod p * q$  を求めるためのべき乗剰余計算方法であって、前記データ  $C$  の  $p$  による剰余値  $C_p$  の剰余系表現及び前記パラメータ  $d$  の  $(p - 1)$  による剰余値  $d_p$  に基づき、 $(C_p^{d_p}) * B \bmod p$  又はこれに  $p$  が加えられた値の剰余系表現を求めるとともに、前記データ  $C$  の  $q$  による剰余値  $C_q$  の剰余系表現及び前記パラメータ  $d$  の  $(q - 1)$  による剰余値  $d_q$  に基づき、 $(C_q^{d_q}) * B \bmod q$  又はこれに  $q$  が加えられた値の剰余系表現を求め、求められた両剰余系表現に基づき、法  $p * q$  の元で  $C^d$  と合同である整数



$m'$  の剰余系表現を求め、求められた前記剰余系表現を 2 進数表現に変換して得られた前記整数  $m'$  の値に基づき、前記計算結果  $m$  を求めることを特徴とする。

#### 【0031】

なお、装置に係る本発明は方法に係る発明としても成立し、方法に係る本発明は装置に係る発明としても成立する。

また、装置または方法に係る本発明は、コンピュータに当該発明に相当する手順を実行させるための（あるいはコンピュータを当該発明に相当する手段として機能させるための、あるいはコンピュータに当該発明に相当する機能を実現させるための）プログラムとしても成立し、該プログラムを記録したコンピュータ読取り可能な記録媒体としても成立する。

#### 【0032】

本発明によれば、中国剰余定理を利用した演算と剰余系を利用した演算とモンゴメリ演算とを融合させることによって、べき乗剰余計算をより効率良く実行することができる。例えば、剰余系表現を利用したべき乗剰余演算装置にて RSA 暗号の復号変換に中国剰余定理を利用した効率の良い実装が可能となり、同復号変換を約  $1/8$  の処理量に削減できる。

#### 【0033】

##### 【発明の実施の形態】

以下、図面を参照しながら発明の実施の形態を説明する。

#### 【0034】

図 1 に、本発明の一実施形態に係る計算装置の機能的な構成図を示す。

#### 【0035】

本計算装置 1 は、RNS 表現された整数を演算する RNS 演算器 12、2 進数表現での補助的な演算を行う演算部と、外部との入出力を行うための入出力部 11 と、全体を制御する制御部 13 で構成される。

RNS 演算器 12 には、RNS 逆元計算部 (INV) 122、RNS モンゴメリ乗算部 (MM) 123、RNS モンゴメリべき乗部 (MEXP) 124、RNS 乗算部 (MUL) 125、RNS 加算部 (ADD) 126、第 1 の表現変換部 (2 進数表現 → RNS 表現変換部) 127、第 2 の表現変換部 (RNS 表現変換

部→2進数表現) 128、記憶部121が存在する。

2進数表現での補助的な演算部としては、剰余計算部129と加減算部130が存在する。

以上の各演算部のうち規模面で大半はRNS演算器12である。

#### 【0036】

記憶部121は、例えばRNS表現で利用する基底や事前に計算し装置内に記憶されるパラメータ等を記憶する。

#### 【0037】

RNSモンゴメリ乗算部123は、先に説明したRNSモンゴメリ乗算(MM)を行う。

RNSモンゴメリべき乗部124は、先に説明したRNSモンゴメリべき乗(MEXP)を行う。

RNS乗算部125は、先に説明したRNS乗算(MUL)を行う。

RNS加算126は、先に説明したRNS加算(ADD)を行う。

第1の表現変換部127は、2進数表現からRNS表現への変換を行う。

第2の表現変換部128は、RNS表現から2進数表現への変換を行う。

なお、これらについては、例えば、文献「小池、佐野、川村，“Cox-Rowerアーキテクチャによる高速な並列モンゴメリ乗算法”，SCIS2000，B22，2000」に詳しい。

#### 【0038】

RNS逆元計算部(INV)122は、 $\langle x \rangle_a$ を入力として $\langle -x^{-1} \rangle_a$ を計算する。すなわち、 $\langle x \rangle_a$ の各基底 $a_i$ および要素 $x_i$ について、 $x_i \pmod{a_i}$ から、 $-x_i^{-1} \pmod{a_i}$ を計算する。具体的には、例えば次の手順で実行する。

＜基底 $a_i$ における逆元計算＞

(Step0) 基底 $a_i$ に対するCarmichael関数 $\lambda(a_i)$ を計算し、記憶部(ROM)に保存しておく。Carmichael関数 $\lambda$ の具体的な計算式は例えば“岡本龍明、山本博資著、「現代暗号」、産業図書、p.16”に示されている。 $\lambda(a_i)$ のビットサイズは $a_i$ のビットサイズ以下である。

法  $a_i$  と互いに素な全ての  $x$  ( $< a_i$ ) に対して、 $x^{\lambda(a_i)} = 1 \pmod{a_i}$  になる。ここでは入力  $x$  として RSA 暗号の秘密鍵  $p, q$  (素数) もしくはその積  $N$  (2 素数の積) を想定しているため、これらは必ず法  $a_i$  と互いに素になる。

(Step1)  $x_i^{-1} = x_i^{(\lambda(a_i) - 1)} \pmod{a_i}$  を演算ユニットでの剰余乗算により計算する。

(Step2)  $-x_i^{-1} = a_i - x_i^{-1}$  により計算する。

以上の計算のうち、(Step1) は Carmichael 関数  $\lambda(a_i)$  のビットサイズが  $a_i$  のビットサイズ以下であるため、演算ユニットのワード数を 32 ビットとすると、64 回以下のワードサイズの剰余乗算に相当する。

#### 【0039】

剰余計算部 129 は、2 進数表現の被除数  $x$  と除数  $y$  を入力して、 $x \pmod{y}$  を計算する。この計算手順は通常の除算で実行可能であり、例えば “Knuth 著、中川圭介訳、「準数値算法／算術演算」、サイエンス社、p. 90” に示されている。概ね、 $x_1 * x_2$  と同じ計算量である。

加減算部 130 は、2 進数の加減算等を行う。

本計算装置 1 は、次の RNS 演算を組み合わせる CRT べき乗を実行する。

・RNS モンゴメリ乗算  $\langle z \rangle = MM(\langle x \rangle_{aUb}, \langle y \rangle_{aUb}, p, aUb)$

ここで、 $z = x * y * A^{-1} \pmod{p}$  or  $z = (x * y * A^{-1} \pmod{p}) + p$  に相当する。

・RNS モンゴメリべき乗  $\langle z \rangle = MEXP(\langle x \rangle_{aUb}, e, p, aUb)$

ここで、 $z = x^e * A^{-(e-1)} \pmod{p}$  or  $z = (x^e * A^{-(e-1)} \pmod{p}) + p$  に相当する。

・RNS 乗算  $\langle z \rangle = MUL(\langle x \rangle_{aUb}, \langle y \rangle_{aUb}, a)$

ここで、 $z = x * y \pmod{A}$  (基底  $a$  での  $x$  と  $y$  の乗算) に相当する。

・RNS 加算  $\langle z \rangle = ADD(\langle x \rangle_{aUb}, \langle y \rangle_{aUb}, a)$

ここで、 $z = x + y \pmod{A}$  (基底  $a$  での  $x$  と  $y$  の加算) に相当する。

## 【0040】

上記RNS演算における最後の引数（ $a$ や $a \cup b$ など）は、RNS表現で利用される基底を表す。基底 $a$ の要素の積の値を $A$ 、基底 $b$ の要素の積の値を $B$ とすると、基底 $a \cup b$ の要素の積の値は $AB$ になる。RNSモンゴメリ乗算とRNSモンゴメリべき乗の出力は $z \leq A, B$ となる。

## 【0041】

上記のようにRNSモンゴメリ乗算およびRNSモンゴメリべき乗においては、モンゴメリ乗算の性質から、モジュラス $p$ の値だけ結果が大きい場合がある。すなわち、 $MM(\langle x \rangle, \langle y \rangle, p, a \cup b) < 2p$ である。法 $p$ を固定した場合、RNSモンゴメリ乗算とRNSモンゴメリべき乗の出力はいずれも $2p$ 未満であるが、これらの出力はそのままRNSモンゴメリ乗算およびRNSモンゴメリべき乗に入力可能である。

## 【0042】

本計算装置1内には、次のパラメータは、予め記憶しておくものとする。

事前登録パラメータ：基底 $a$ 、基底 $b$ 、基底 $a$ の要素の積 $A$ 、基底 $b$ の要素の積 $B$ 、基底 $a$ および基底 $b$ の全要素の積 $AB$ 、 $B^2$

なお、基底 $a$ 、 $b$ とCRTべき乗におけるパラメータサイズの関係として、少なくとも、 $p, q \leq A$ 、かつ、 $p, q \leq B$ が必要である。この結果、 $N = p * q$ に対し、少なくとも $N \leq AB$ となる。

## 【0043】

CRTべき乗を実行するために外部から本計算装置1へ入力するパラメータは、ここでは、以下の通りとする。

外部入力パラメータ：暗号文 $C$ 、 $d_p = d \bmod (p-1)$ 、 $d_q = d \bmod (q-1)$ 、 $N (= p * q)$ 、 $p$ 、 $q$ 、 $p$ の法 $q$ における逆元 $p_{inv} = p^{-1} \bmod q$ 、 $q$ の法 $p$ における逆元 $q_{inv} = q^{-1} \bmod p$

図2に、本計算装置1におけるCRTべき乗の処理手順の一例を示す。また、図3に、本計算装置1の各演算部に関する内部構成例を示す。

## 【0044】

(ステップS0) : 外部入力パラメータC、dp、dq、N、p、q、pinv、qinvを入力する。

【0045】

以下の手順のうち、ステップS1-p～S9-pと、ステップS1-q～S9-qとでは、いずれの対応するステップSi-pとステップSi-qにおいても、Nの2つの素因数であるpとqに関する同様の演算を実行している。

【0046】

(ステップS1-p)

第1の表現変換部127を利用して、2進数表現pを、基底aUbによるRNS表現 $\langle p \rangle$  ( $= \langle p \rangle_a \cup \langle p \rangle_b = \{p \bmod a_1, p \bmod a_2, \dots, p \bmod a_n\} \cup \{p \bmod b_1, p \bmod b_2, \dots, p \bmod b_n\}$ )に変換する。

・ 【0047】

(ステップS1-q)

第1の表現変換部127を利用して、2進数表現qを、基底aUbによるRNS表現 $\langle q \rangle$  ( $= \langle q \rangle_a \cup \langle q \rangle_b = \{q \bmod a_1, q \bmod a_2, \dots, q \bmod a_n\} \cup \{q \bmod b_1, q \bmod b_2, \dots, q \bmod b_n\}$ )に変換する。

【0048】

(ステップS2-p)

RNS逆元計算部122を利用して、(ステップS1-pで求められた) $\langle p \rangle_b$ から、 $\langle -p^{-1} \rangle_b$ を計算する。

【0049】

(ステップS2-q)

RNS逆元計算部122を利用して、(ステップS1-qで求められた) $\langle q \rangle_b$ から、 $\langle -q^{-1} \rangle_b$ を計算する。

【0050】

(ステップS3-p)

剰余計算部129を利用して、 $b_p = B^2 \bmod p$ を計算し、第1の表

現変換部 1 2 7 を利用して、 $b_p$  を 2 進数表現から基底  $a U b$  による RNS 表現  $\langle b_p \rangle$  に変換する。

【0 0 5 1】

(ステップ S 3 - q)

剰余計算部 1 2 9 を利用して、 $b_q = B^2 \bmod q$  を計算し、第 1 の表現変換部 1 2 7 を利用して、 $b_q$  を 2 進数表現から基底  $a U b$  による RNS 表現  $\langle b_q \rangle$  に変換する。

【0 0 5 2】

(ステップ S 4 - p)

第 1 の表現変換部 1 2 7 を利用して、 $p_{inv}$  を 2 進数表現から基底  $a U b$  による RNS 表現  $\langle p_{inv} \rangle$  に変換する。

【0 0 5 3】

(ステップ S 4 - q)

第 1 の表現変換部 1 2 7 を利用して、 $q_{inv}$  を 2 進数表現から基底  $a U b$  による RNS 表現  $\langle q_{inv} \rangle$  に変換する。

【0 0 5 4】

(ステップ S 5 - p)

剰余計算部 1 2 9 を利用して、 $C_p = C \bmod p$  を計算し、第 1 の表現変換部 1 2 7 を利用して、 $C_p$  を 2 進数表現から基底  $a U b$  による RNS 表現  $\langle C_p \rangle$  に変換する。

【0 0 5 5】

(ステップ S 5 - q)

剰余計算部 1 2 9 を利用して、 $C_q = C \bmod q$  を計算し、第 1 の表現変換部 1 2 7 を利用して、 $C_q$  を 2 進数表現から基底  $a U b$  による RNS 表現  $\langle C_q \rangle$  に変換する。

【0 0 5 6】

(ステップ S 6 - p)

RNS モンゴメリ乗算部 1 2 3 を利用して、 $\langle C_p' \rangle = MM(\langle C_p \rangle, \langle b_p \rangle, p, a U b)$  を計算する。

＜先に説明したアルゴリズムを利用する場合の処理内容＞

Step-M-1 :  $\langle s \rangle_a = \langle Cp \rangle_a * \langle bp \rangle_a$  を計算する。

Step-M-2 :  $\langle s \rangle_b = \langle Cp \rangle_b * \langle bp \rangle_b$  を計算する。

Step-M-3 :  $\langle t \rangle_b = \langle s \rangle_b * \langle (-p^{(-1)}) \rangle_b$  を計算する。

Step-M-4 :  $\langle t \rangle_b$  を  $\langle t \rangle_a$  へ基底変換する。

Step-M-5 :  $\langle u \rangle_a = \langle t \rangle_a * \langle p \rangle_a$  を計算する。

Step-M-6 :  $\langle v \rangle_a = \langle s \rangle_a + \langle u \rangle_a$  を計算する。

Step-M-7 :  $\langle Cp' \rangle_a = \langle v \rangle_a * \langle B^{(-1)} \rangle_a$  を計算する。

Step-M-8 :  $\langle Cp' \rangle_a$  を  $\langle Cp' \rangle_b$  へ基底変換する。

これにより、 $Cp' = C * B \mod p$  または  $Cp' = (C * B \mod p) + p$  のいずれかに対応する RNS 表現  $\langle Cp' \rangle$  が求められる。

【0057】

(ステップ S6-q)

RNS モンゴメリ乗算部 123 を利用して、 $\langle Cq' \rangle = MM(\langle Cq \rangle, \langle bq \rangle, q, a \cup b)$  を計算する。なお、先に説明したアルゴリズムを利用する場合の処理内容は、ステップ S6-p の処理内容において、p と q を入れ替えたものである。

これにより、 $Cq' = C * B \mod q$  または  $Cq' = (C * B \mod q) + q$  のいずれかに対応する RNS 表現  $\langle Cp' \rangle$  が求められる。

【0058】

(ステップ S7-p)

RNS モンゴメリべき乗部 124 を利用して、 $\langle mp' \rangle = MEXP(\langle Cp' \rangle, dp, p, a \cup b)$  を計算する。

＜先に説明したアルゴリズムを利用する場合の処理内容＞

Step-E-1 :  $i = k$  とする。 $\langle y \rangle_a \cup b = \langle B \rangle_a \cup b$  とする。

Step-E-2 :  $\langle y \rangle_a \cup b = MM(\langle y \rangle_a \cup b, \langle y \rangle_a \cup b, N, a \cup b)$  を計算する。

Step-E-3 :  $d p\_i = 1$  ならば、 $\langle y \rangle\_a U b = MM(\langle y \rangle\_a U b, \langle C p' \rangle\_a U b, p, a U b)$  を計算する。 $d p\_i \neq 1$  ならば、なにもしない (nop)。

ここで、 $d p\_i$  は、 $d p$  の2進表現 ( $d p\_k, d p\_k-1, \dots, d p\_1$ ) における下位から  $i$  ビット目の値である。

Step-E-4 :  $i = i - 1$  とする。

Step-E-5 :  $i = 0$  ならば、終了する。 $i \neq 0$  ならば、Step-E-2 へ戻る。

これにより、 $m p' = (C p^{d p}) * B \bmod p$  または  $m p' = ((C p^{d p}) * B \bmod p) + p$  のいずれかに対応するRNS表現  $\langle m p' \rangle$  が求められる。

【0059】

(ステップS7-q)

RNSモンゴメリべき乗部124を利用して、 $\langle m q' \rangle = MEXP(\langle C q' \rangle, d q, q, a U b)$  を計算する。なお、先に説明したアルゴリズムを利用する場合の処理内容は、ステップS7-pの処理内容において、 $p$  と  $q$  を入れ替えたものである。

これにより、 $m q' = (C q^{d q}) * B \bmod q$  または  $m q' = ((C q^{d q}) * B \bmod q) + q$  のいずれかに対応するRNS表現  $\langle m q' \rangle$  が求められる。

【0060】

(ステップS8-p)

RNSモンゴメリ乗算部123を利用して、 $\langle t p \rangle = MM(\langle m p' \rangle, \langle q^{(-1) \bmod p} \rangle, p, a U b)$  を計算する。

<先に説明したアルゴリズムを利用する場合の処理内容>

Step-M-1 :  $\langle s \rangle\_a = \langle m p' \rangle\_a * \langle q_{inv} \rangle\_a$  を計算する。

Step-M-2 :  $\langle s \rangle\_b = \langle m p' \rangle\_b * \langle q_{inv} \rangle\_b$  を計算する。

Step-M-3 :  $\langle t \rangle\_b = \langle s \rangle\_b * \langle (-p^{(-1)}) \rangle\_b$  を計算する。

Step-M-4 :  $\langle t \rangle\_b$  を  $\langle t \rangle\_a$  へ基底変換する。



Step-M-5 :  $\langle u \rangle_a = \langle t \rangle_a * \langle p \rangle_a$  を計算する。

Step-M-6 :  $\langle v \rangle_a = \langle s \rangle_a + \langle u \rangle_a$  を計算する。

Step-M-7 :  $\langle tp \rangle_a = \langle v \rangle_a * \langle B^{(-1)} \rangle_a$  を計算する。

Step-M-8 :  $\langle tp \rangle_a$  を  $\langle tp \rangle_b$  へ基底変換する。

これにより、 $tp = (Cp^{dp})q^{(-1)} \bmod p$  または  $tp = ((Cp^{dp})q^{(-1)} \bmod p) + p$  のいずれかに対応するRNS表現  $\langle tp \rangle$  が求められる。

【0061】

(ステップS8-q)

RNSモンゴメリ乗算部123を利用して、 $\langle tq \rangle = MM(\langle mq' \rangle, \langle p^{(-1)} \bmod q \rangle, p, a \cup b)$  を計算する。なお、先に説明したアルゴリズムを利用する場合の処理内容は、ステップS8-pの処理内容において、pとqを入れ替えたものである。

これにより、 $tq = (Cq^{dq})p^{(-1)} \bmod q$  または  $tq = ((Cq^{dq})p^{(-1)} \bmod q) + q$  のいずれかに対応するRNS表現  $\langle tq \rangle$  が求められる。

【0062】

(ステップS9-p)

RNS乗算部125を利用して、 $\langle up \rangle = MUL(\langle tp \rangle, \langle q \rangle, a \cup b)$  を計算する。

これにより、 $up = tp * q \bmod AB$  に対応するRNS表現  $\langle up \rangle$  が求められる。

【0063】

(ステップS9-q)

RNS乗算部125を利用して、 $\langle uq \rangle = MUL(\langle tq \rangle, \langle p \rangle, a \cup b)$  を計算する。

これにより、 $uq = tq * p \bmod AB$  に対応するRNS表現  $\langle uq \rangle$  が求められる。

【0064】

(ステップS10)

RNS加算126を利用して、 $\langle m' \rangle = \text{ADD}(\langle u_p \rangle, \langle u_q \rangle, aUb)$ を計算する。

これにより、 $m' = u_p + u_q \bmod AB$ に対応するRNS表現 $\langle m' \rangle$ が求められる。

【0065】

(ステップS11)

第2の表現変換部128を利用して、 $\langle m' \rangle$ をRNS表現(基底 $aUb$ )から2進数表現 $m'$ に変換する。

ここで、 $m'$ は $N$ 以上である場合があるので、加減算部130は、 $m'$ が $N$ 以上であれば、これを $N$ 未満の値にするための処理を行う。

(ステップS12)

$m'$ を $m$ にコピーする(待避させる)。

(ステップS13)

$m' = m' - N$ を計算する。

(ステップS14)

$m' < 0$ であるか否か判定する。 $m' < 0$ でなければ、ステップS12へ戻る。  
 $m' < 0$ であるならば、ループを抜けて、ステップS15へ移る。

(ステップS15)

$m$ を出力して、終了する。

なお、ステップS12～ステップS15は、例えば図4のステップS21～ステップS24など、他の手順でもかまわない。

【0066】

また、 $N$ は、外部から入力せずに、加減算部130が $p * q$ により求めるようにしても良い。

【0067】

上記手順において、ステップS5-p, S6-pおよびステップS5-q, S6-qでは、 $C_p' = C * B \bmod p (+p)$ および $C_q' = C * B \bmod q (+q)$ を計算しており、先に説明した通常のCRTべき乗での(Step-C

-2) の処理に対応している。

ステップ S7-p およびステップ S7-q は、通常の CRT べき乗での (Step-C-3) の処理に対応している。

ステップ S8-p、S9-p、S8-q、S9-q、S10 は、先に説明した通常の CRT べき乗での (Step-C-4) の処理に対応している。ここでは、(Step-C-4) の処理が次のように変形できることを利用している。

$$\begin{aligned} m &= m_p * (q^{(-1) \bmod p}) * q + m_q * (p^{(-1) \bmod q}) * p \quad (\bmod N) \\ &= \{m_p * (q^{(-1) \bmod p}) \bmod p\} * q + \{m_q * (p^{(-1) \bmod q}) \bmod q\} * p \quad (\bmod N) \end{aligned}$$

ステップ S11 の結果である  $m'$  は、RNS モンゴメリ乗算での  $p$  および  $q$  の加算誤差と、加算誤差が無いとしても CRT べき乗剰余計算では  $m' < 2N$  であることから、加算誤差を考慮すると  $m' < 4N$  となる。従って、最大で  $m'$  から  $3N$  を引く必要があり、このために必要な補正をステップ S12 からステップ S14 で行っている。 $m'$  は 2 進数に変換してあるので正負の判定は容易である。この処理が積に説明した通常の CRT べき乗での (Step-C-4) の処理における法  $N$  での剰余値を求める手順に相当する。

【0068】

本 CRT べき乗剰余計算での各計算ステップは RNS 演算器で実行可能な演算機能を用いて実行可能である。特にステップ S7-p とステップ S7-q の RNS モンゴメリべき乗が計算処理の大半を占めるが、これらは法  $p$ 、 $q$  よりわずかに大きな基底  $A$ 、 $B$  を用いた和集合  $A \cup B$  を基底として利用できることが重要である。

RNS モンゴメリ乗算の計算量は、その内部で実行する基底変換の計算量で評価できる。この処理では、1 つの基底要素について見た場合にワードサイズでの乗算を基底サイズ  $n$  のオーダだけ必要とし、さらにこれを変換元の基底における全ての基底要素について実行する。従って、RNS モンゴメリ乗算の計算量は、基底サイズ  $n$  の 2 乗のオーダとなる。また、RNS モンゴメリべき乗の計算量は、RNS モンゴメリ乗算を指数のビットサイズ  $L_e$  だけ繰り返す処理に相当す

る。よって、RNSモンゴメリべき乗の計算量は $O(n^2 * L_e)$ となる。

具体的に、例えば、1024ビットのRSA暗号を想定する。この場合、秘密鍵 $d$ 、 $N$ 、暗号文 $C$ はいずれも1024ビットになる。従って、これを、従来のようにRNS表現でのモンゴメリべき乗で実行する場合、利用する基底 $a'$ （および $b'$ ）は最低でも要素数33（ $=1024/32$ （ワードサイズ） $+1$ ）となる。一方、本実施形態のようにCRTべき乗で利用する秘密鍵 $d_p$ 、 $d_q$ 、 $p$ 、 $q$ 、 $C$ を法 $p$ 、 $q$ で縮小した値 $C_p$ 、 $C_q$ はいずれも512ビットであるため、利用する基底 $a$ （および $b$ ）は最低で要素数17（ $=512/32$ （ワードサイズ） $+1$ ）となる。最低の基底要素数を利用するのが処理時間の面で最も効率的であるので、この前提でCRTによるべき乗剰余計算とCRTによらないべき乗剰余計算の計算量を比較してみる。RNSモンゴメリ乗算については、その計算量は、CRTを用いる場合は、CRTを用いない場合の $1/4$ となる。べき指数のサイズについては、CRTを用いる場合にはCRTを用いない場合の $1/2$ になるが、CRTを用いる場合にはRNSモンゴメリべき乗を2回計算する必要がある。よって、全体としては、本CRTべき乗剰余計算によれば、従来のRNSモンゴメリべき乗に比べて、約 $1/4$ の処理量でRSA復号変換が実現できる。また、RNSモンゴメリべき乗を2つの回路で同時に実行すれば、従来のRNSモンゴメリべき乗に比べて、約 $1/8$ の処理量でRSA復号変換が実現できる。

#### 【0069】

以下では、本実施形態のバリエーションについて説明する。

#### 【0070】

図2の手順のうち、ステップ $S1-p \sim S5-p$ の手順は、ステップ $S2-p$ がステップ $S1-p$ の後になる制約がある以外、どのような順番で行ってもかまわない（剰余計算部129や表現変換部127を並列処理可能にして、それらの全部または一部を並列処理してもよい）。

#### 【0071】

また、図2の手順のうち、ステップ $S1-p \sim S9-p$ と、ステップ $S1-q \sim S9-q$ とでは、いずれも対応するステップ $S_i-p$ とステップ $S_i-q$ で $N$

の2つの素因数である  $p$  と  $q$  に関する同様の演算を実行している。ステップ  $S'1 - p \sim S9 - p$ 、 $S1 - q \sim S9 - q$  の演算は、 $p$  パートと  $q$  パートを逐次的に行っても良いし、全ての  $p$  パートを実行後に全ての  $q$  パートを実行するようにしても良い。後者の方が、中間変数のメモリへの退避・復旧が減る分、効率が良くなる可能性がある。

また、 $p$  パートと  $q$  パートをパイプライン的に処理することも可能である。

また、該当する演算部の全部または一部を並列処理可能にして  $p$  パートと  $q$  パートとを並列的に実行することも可能である。 $p$  パートと  $q$  パートとを別々に記述した場合の計算装置1の各演算部に関する内部構成例を図5に示す。

また、例えば、RNS モンゴメリ乗算部123、RNS モンゴメリべき乗部124、RNS 乗算部125、RNS 加算部126の全部、あるいはそれらのうちRNS モンゴメリ乗算部123およびRNS モンゴメリべき乗部124のみ、あるいはそれらのうちRNS モンゴメリべき乗部124のみを、並列処理可能にして  $p$  パートと  $q$  パートとを並列的に実行することも可能である。

もちろん、各演算部は、RNS 演算に由来する並列計算を行って高速化を図ることが可能である。この場合に、基底の全ての要素に対する演算を同時に実行するように構成することも可能であるし、基底の一部（例えば、基底サイズの整数分の一に相当する個数）の要素に対する演算を同時に実行するように構成することも可能である。

#### 【0072】

また、これまで説明した構成例では、 $p \text{ inv} = p^{-1} \bmod q$ 、 $q \text{ inv} = q^{-1} \bmod p$  を外部から入力する例を示したが、それらを  $p$ 、 $q$  から計算するようにしてもよい。この場合には、図5に示すように、2進数表現での補助的な演算部として、剰余計算部129と加減算部130の他に、さらに逆元計算部131を設ければよい。

逆元計算部131では、2進数表現の整数  $x$  と法の値  $y$  を入力して、 $x^{-1} \bmod y$  を計算する。この計算は拡張ユークリッドの互除法と呼ばれるアルゴリズムで実行されることが多い。例えば、“Knuth 著、中川圭介訳、「準数値算法／算術演算」、サイエンス社、p. 162”に示されている。概ね

、 $y$ のサイズの剰余乗算10回分程度の計算量である。

【0073】

また、これまで説明した構成例では、 $dp = d \bmod (p-1)$ 、 $dq = d \bmod (q-1)$ を外部から入力する例を示したが、それらを $p$ 、 $q$ から計算するようにしてもよい。この計算は、剰余計算部129により行うことができる。

【0074】

$p_{inv}$ 、 $q_{inv}$ 、 $dp$ 、 $dq$ を $p$ 、 $q$ から計算するようにした場合の計算装置1の各演算部に関する内部構成例を図6に示す。

【0075】

また、外部入力パラメータ（暗号文 $C$ 、 $dp = d \bmod (p-1)$ 、 $dq = d \bmod (q-1)$ 、 $N (= p * q)$ 、 $p$ 、 $q$ 、 $p_{inv} = p^{-1} \bmod q$ 、 $q_{inv} = q^{-1} \bmod p$ )のうち暗号文 $C$ 以外のものは、RSAの秘密鍵に相当するパラメータであり、それらの全部または一部を本計算装置1内部に記憶しておくことも可能である。この場合には、外部からは、暗号文 $C$ と本演算装置1内部の鍵パラメータ群を選択するのに必要な鍵識別情報を入力するようにすればよい。

【0076】

また、図2のステップ $S1-p \sim S4-p$ およびステップ $S1-q \sim S4-q$ で示される計算は、RSAの秘密鍵( $p$ 、 $q$ 、 $p_{inv}$ 、 $q_{inv}$ )のみに依存した計算であるが、RSAによる暗号文 $C$ はセッション毎に異なるのに対し、RSA秘密鍵はそれほど変更されないことが多い(RSA秘密鍵が不変のシステムもあり得る)。

そこで、ステップ $S1-p \sim$ ステップ $S4-q$ まで実行した結果を保存しておき、以後同じRSA秘密鍵が用いられる限り、ステップ $S1-p \sim$ ステップ $S4-q$ までスキップし、先に保存しておいた結果を利用してステップ $S5-p$ 以降の処理を行うとともに、RSA秘密鍵が変更されたときに改めてステップ $S1-p \sim$ ステップ $S4-q$ まで実行するようにしてもよい。

また、RSA秘密鍵を鍵識別情報で管理する場合には、鍵識別情報に対応付け

て、上記結果を保存しておくようにしてもよい。

【0077】

また、RSA秘密鍵が唯一不変の場合には、外部からはCのみを入力するものとし、RSA秘密鍵のみに依存するデータ ( $p$ 、 $q$ 、 $N$ 、 $\langle p \rangle$ 、 $\langle q \rangle$ 、 $\langle -p^{(-1)} \rangle_{\text{b}}$ 、 $\langle -q^{(-1)} \rangle_{\text{b}}$ 、 $\langle bp \rangle$ 、 $\langle bq \rangle$ 、 $\langle pinv \rangle$ 、 $\langle qinv \rangle$ 、 $\langle bp \rangle$ 、 $\langle bq \rangle$ ) は、予め記憶部に記憶しておくようにしてもよい。

また、RSA秘密鍵が複数ある場合には、外部からはCおよび鍵識別情報のみを入力するものとし、RSA秘密鍵のみに依存するデータ ( $p$ 、 $q$ 、 $N$ 、 $\langle p \rangle$ 、 $\langle q \rangle$ 、 $\langle -p^{(-1)} \rangle_{\text{b}}$ 、 $\langle -q^{(-1)} \rangle_{\text{b}}$ 、 $\langle bp \rangle$ 、 $\langle bq \rangle$ 、 $\langle pinv \rangle$ 、 $\langle qinv \rangle$ 、 $\langle bp \rangle$ 、 $\langle bq \rangle$ ) を鍵識別情報に対応付けて予め記憶部に記憶しておき、外部から入力された鍵識別情報に対応するものを記憶部から読み出して用いるようにしてもよい。

【0078】

また、2種類の基底を用いる場合に、基底  $a = \{a_1, a_2, \dots, a_{n1}\}$  と基底  $b = \{b_1, b_2, \dots, b_{n2}\}$  について、 $n1 = n2 = n$  として説明したが、 $n1 \neq n2$  とすることも可能である。

【0079】

なお、以上では、本発明を図8のような復号変換に適用した場合について説明したが、暗号変換 ( $C = m^e \bmod N$ ) は復号変換 ( $m = C^d \bmod N$ ) と同様の計算式により表現されるので、もちろん、本発明は暗号変換にも適用可能である (例えば、秘密鍵を持つ装置が、暗号変換するケース)。この場合には、これまでの説明において、暗号文Cの代わりに平文mを入力とし、指数dの代わりに指数eを用いればよい。

【0080】

ここで、本計算装置のハードウェア、ソフトウェア構成について説明する。

本実施の形態では、本計算装置 (復号化装置あるいは暗号化装置) をハードウェアにより実現することを想定して説明したが、ソフトウェアとして実現することも可能である。

ハードウェアとして構成する場合、例えば、半導体装置として形成し、演算ボードあるいは演算カードとして、パーソナル・コンピュータなどの計算機に装備する形態がある。計算機がOSを用いる場合には、この演算デバイス用のドライバをOSに組み込んで用いる形態もある。また、例えば、半導体装置として形成し、AV機器や家電機器等の装置に備えることも可能である。

ソフトウェアで実現する場合、コンピュータに所定の手段を実行させるための（あるいはコンピュータを所定の手段として機能させるための、あるいはコンピュータに所定の機能を実現させるための）プログラムとして実施することができ、また該プログラムを記録したコンピュータ読取り可能な記録媒体として実施することができる。もちろん、マルチプロセッサやパイプライン処理などの種々の高速化技術を利用することも可能である。

#### 【0081】

なお、この発明の実施の形態で例示した構成は一例であって、それ以外の構成を排除する趣旨のものではなく、例示した構成の一部を他のもので置き換えたり、例示した構成の一部を省いたり、例示した構成に別の機能あるいは要素を付加したり、それらを組み合わせたりすることなどによって得られる別の構成も可能である。また、例示した構成と論理的に等価な別の構成、例示した構成と論理的に等価な部分を含む別の構成、例示した構成の要部と論理的に等価な別の構成なども可能である。また、例示した構成と同一もしくは類似の目的を達成する別の構成、例示した構成と同一もしくは類似の効果を奏する別の構成なども可能である。

また、この発明の実施の形態で例示した各種構成部分についての各種バリエーションは、適宜組み合わせて実施することが可能である。

また、この発明の実施の形態は、個別装置としての発明、関連を持つ2以上の装置についての発明、システム全体としての発明、個別装置内部の構成部分についての発明、またはそれらに対応する方法の発明等、種々の観点、段階、概念またはカテゴリに係る発明を包含・内在するものである。

従って、この発明の実施の形態に開示した内容からは、例示した構成に限定されることなく発明を抽出することができるものである。



【 0 0 8 2 】

本発明は、上述した実施の形態に限定されるものではなく、その技術的範囲において種々変形して実施することができる。

【 0 0 8 3 】

【発明の効果】

本発明によれば、中国剰余定理を利用した演算と剰余系を利用した演算とモンゴメリ演算とを融合させることによって、べき乗剰余計算をより効率良く実行することができる。

【図面の簡単な説明】

【図 1】

本発明の実施の形態に係る計算装置の構成例を示す図

【図 2】

同実施形態に係る計算装置の処理手順の一例を示すフローチャート

【図 3】

同実施形態に係る計算装置の各演算部に関する内部構成例を示す図

【図 4】

同実施形態に係る計算装置の処理手順の他の例を示すフローチャート

【図 5】

同実施形態に係る計算装置の各演算部に関する内部構成の他の例を示す図

【図 6】

同実施形態に係る計算装置の他の構成例を示す図

【図 7】

同実施形態に係る計算装置の各演算部に関する内部構成のさらに他の例を示す図

【図 8】

同実施形態に係る計算装置の適用例について説明するための図

【符号の説明】

1 … 計算装置

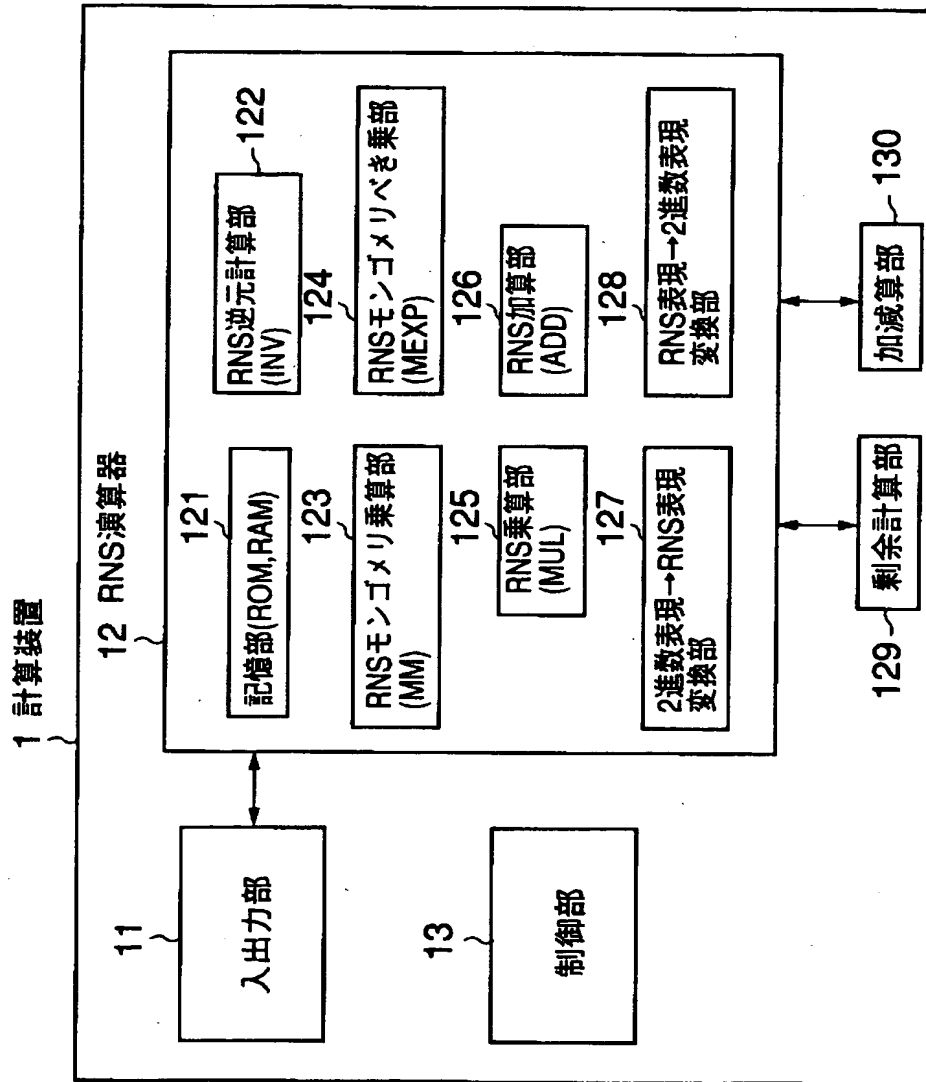
1 1 … 入出力部

- 1 2 … R N S 演算器
- 1 3 … 制御部
- 1 2 1 … 記憶部
- 1 2 2 … R N S 逆元計算部
- 1 2 3 … R N S モンゴメリ乗算部
- 1 2 4 … R N S モンゴメリべき乗部
- 1 2 5 … R N S 乗算部
- 1 2 6 … R N S 加算部
- 1 2 7 … 第 1 の表現変換部
- 1 2 8 … 第 2 の表現変換部
- 1 2 9 … 剰余計算部
- 1 3 0 … 加減算部
- 1 3 1 … 逆元計算部

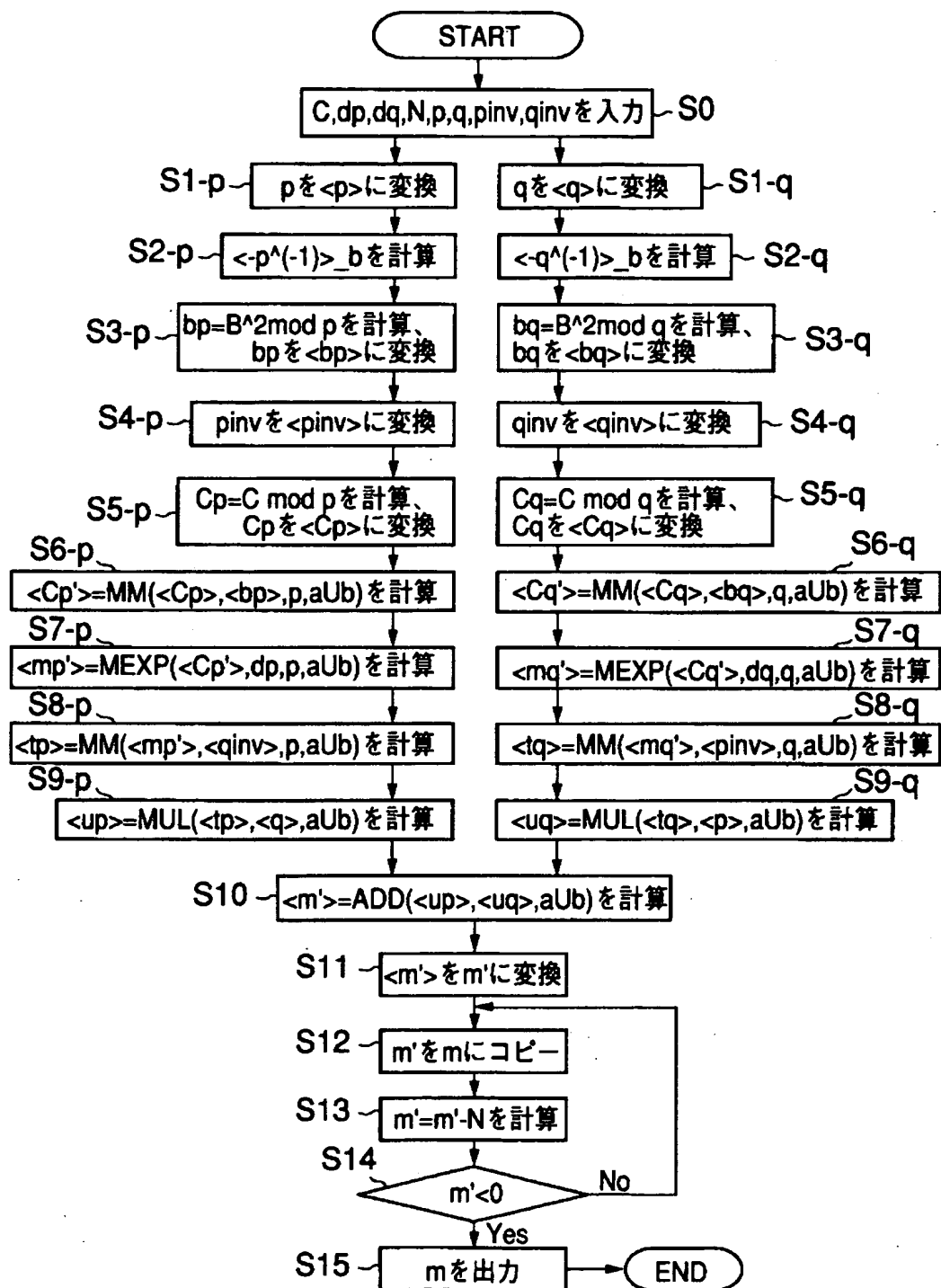
【書類名】

図面

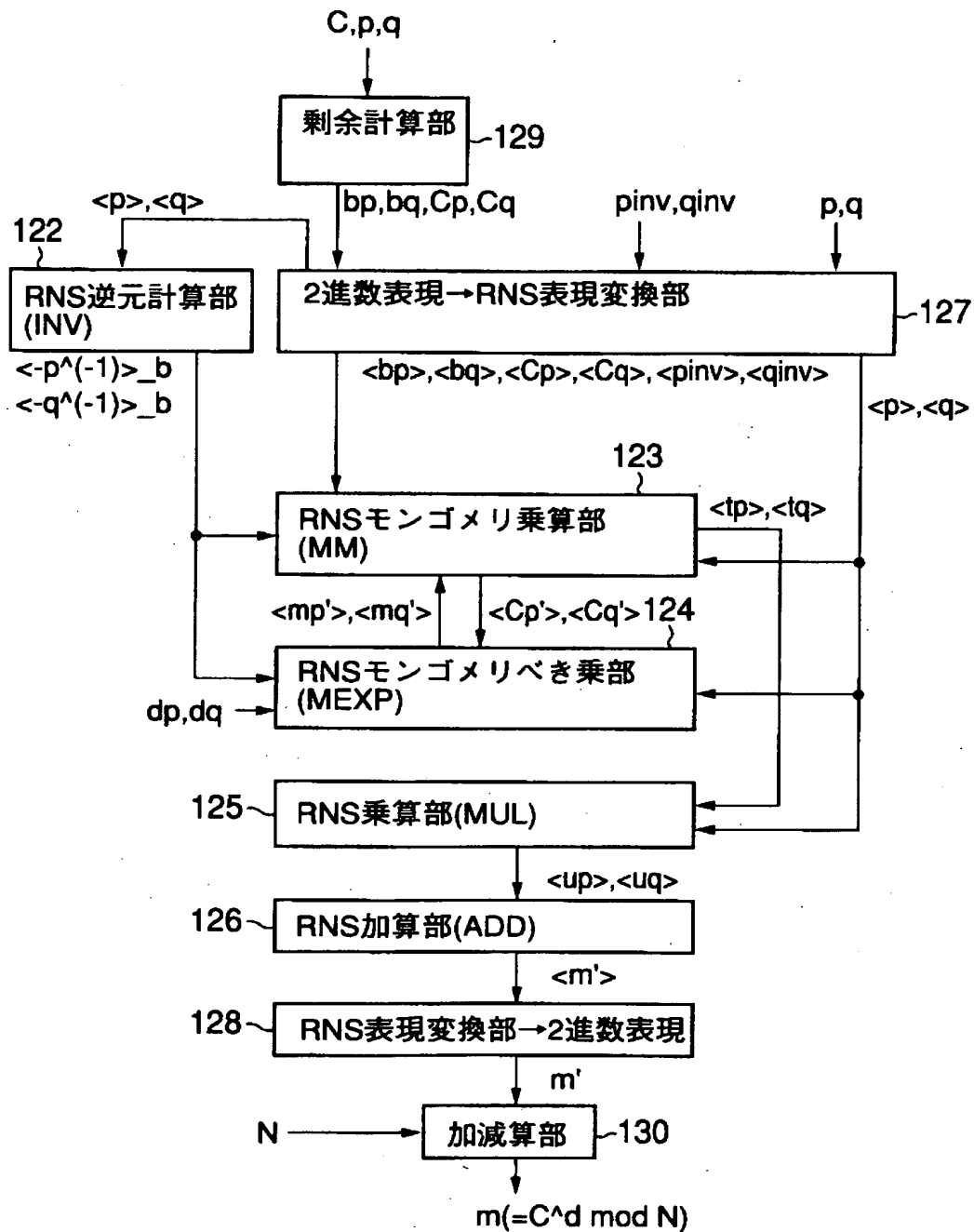
【図 1】



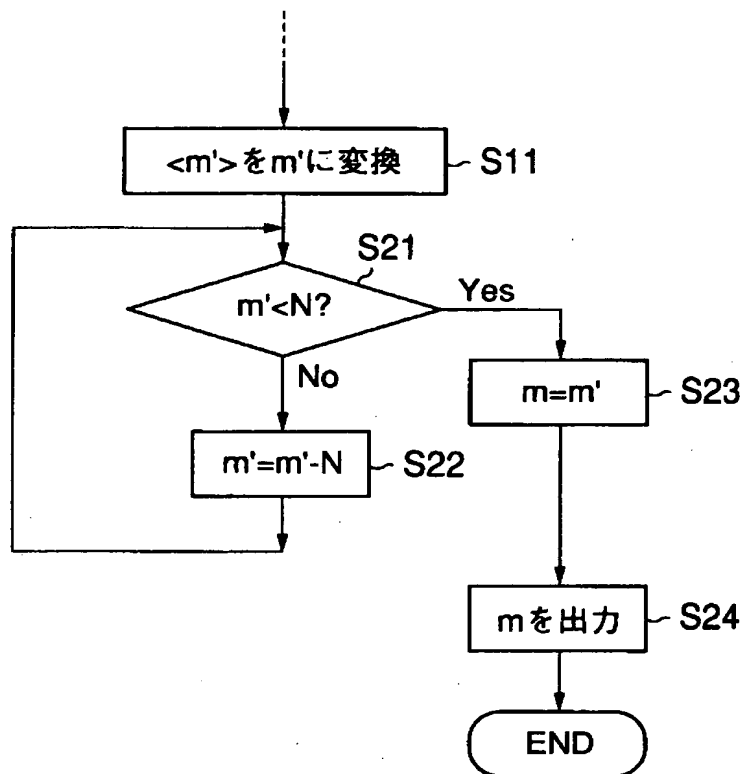
【図 2】



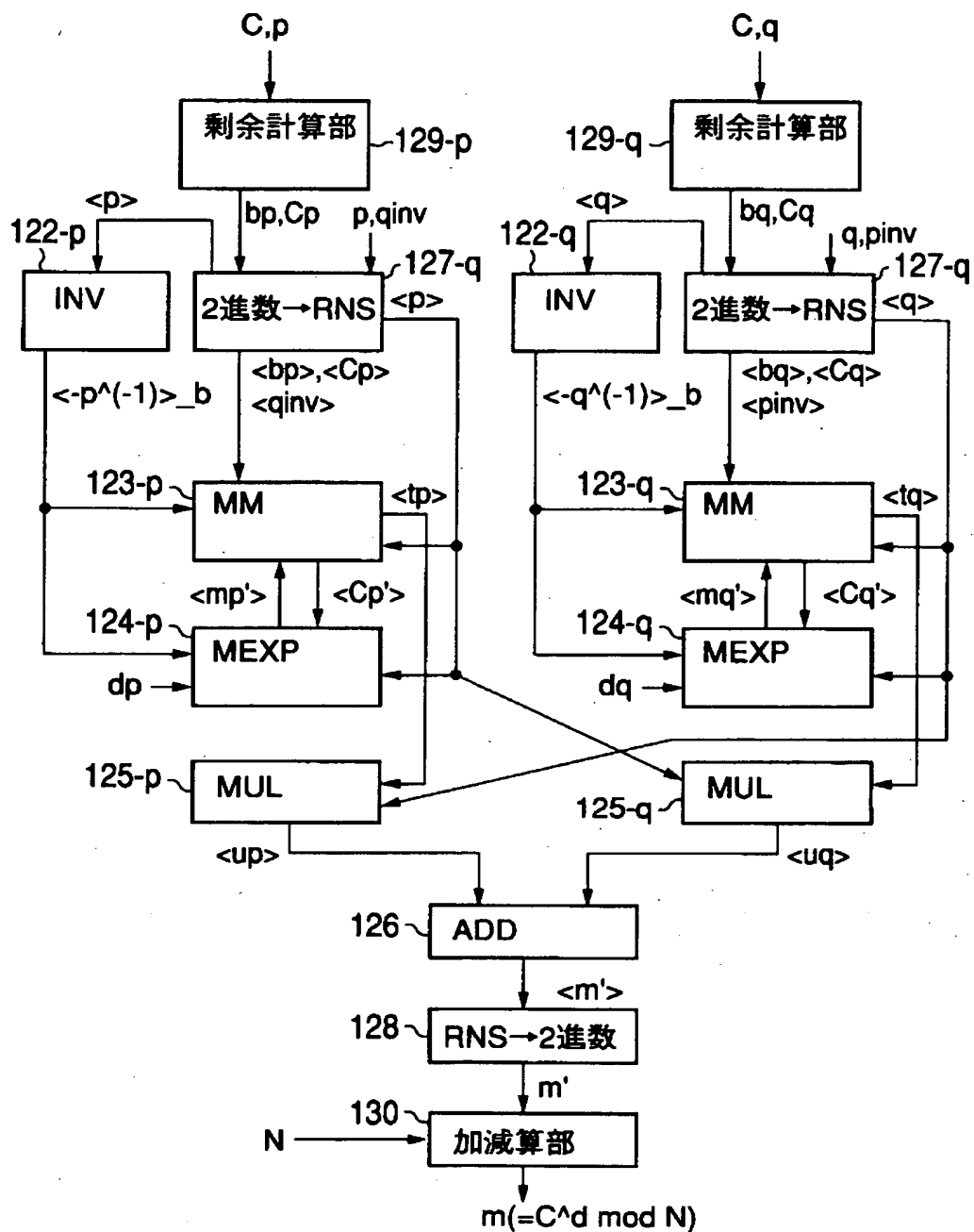
【図 3】



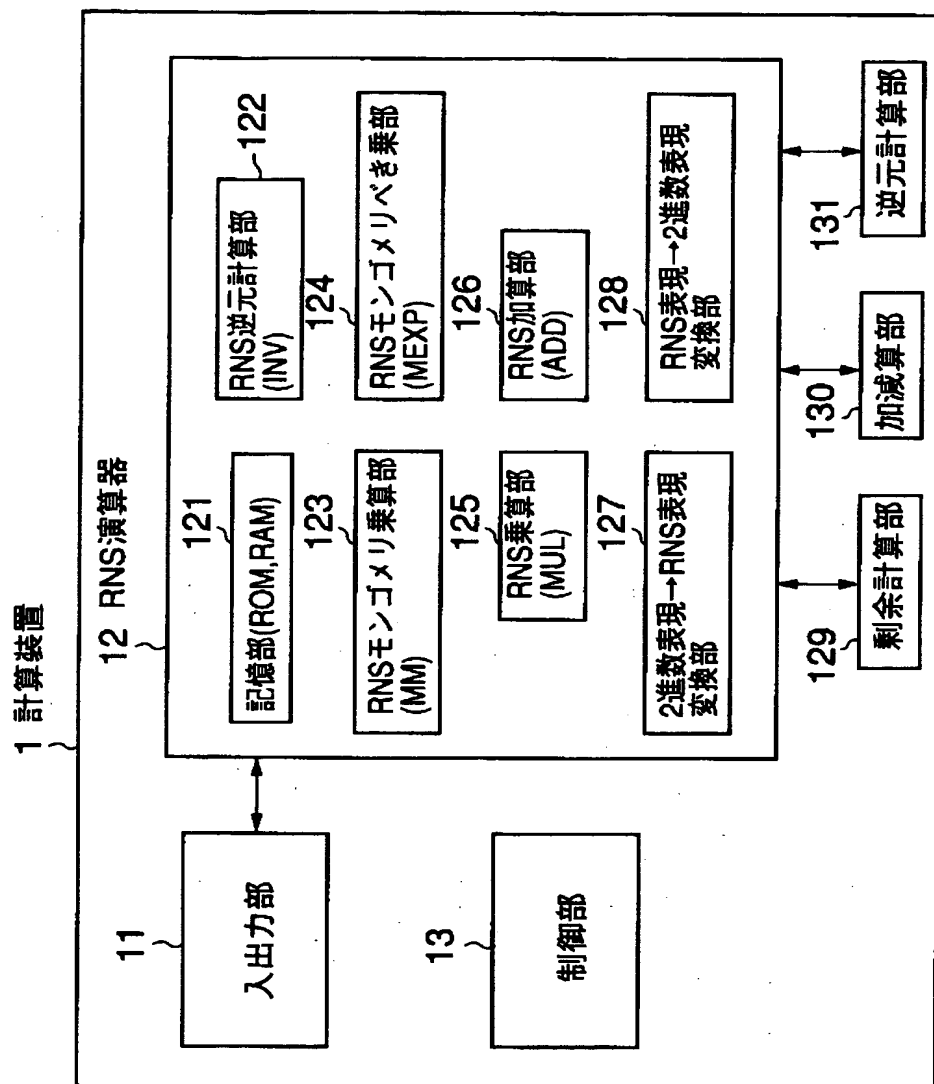
【図 4】



【図 5】

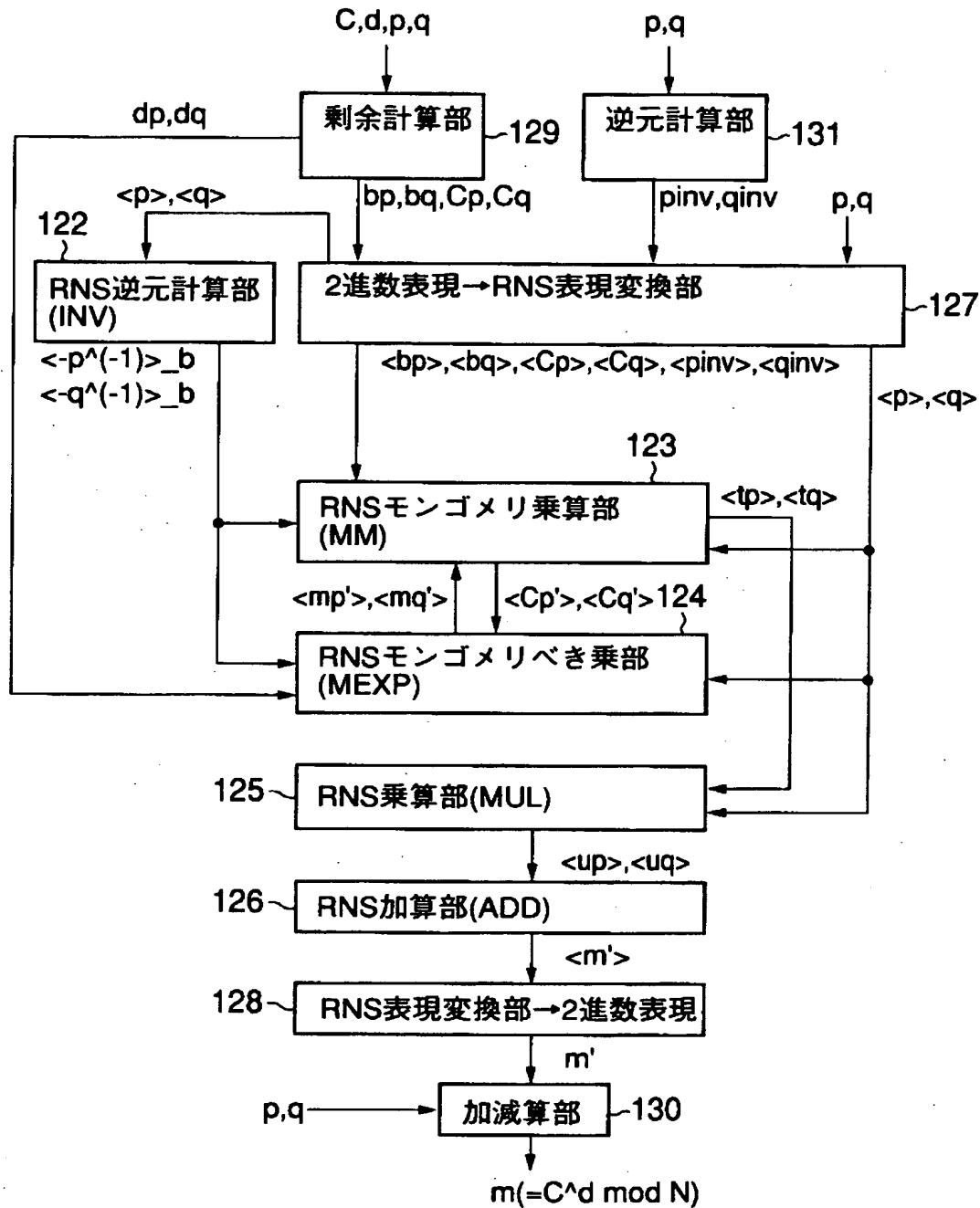


【図 6】

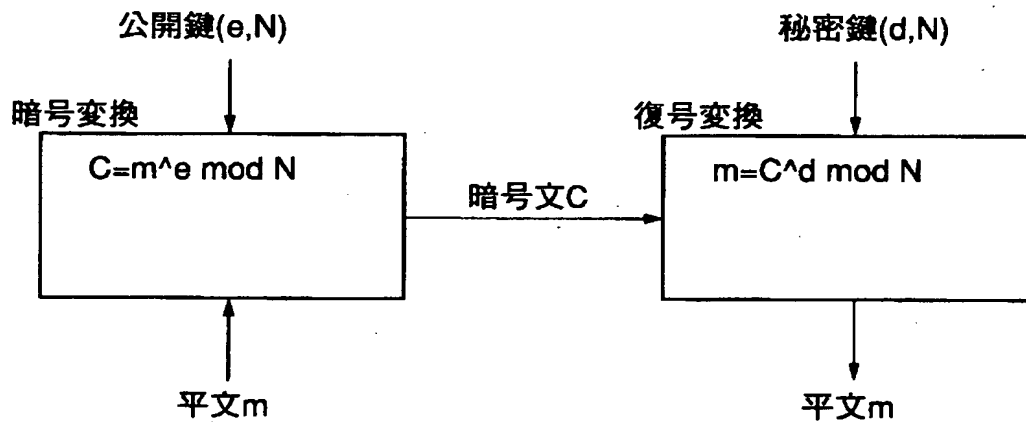




【図 7】



【図 8】



【書類名】 要約書

【要約】

【課題】 より効率良く実行可能にしたべき乗剰余計算装置を提供すること。

【解決手段】 暗号文Cに対し、第1及び第2の基底による剰余系表現を利用して（両基底に含まれる全整数は互いに素、第1の基底の全整数の積A、第2の基底の全整数の積 $B > p, q, A * B > C$ とする）、平文 $m = C^d \bmod p * q$ を求める。まず、pについて $C \bmod p$ の剰余系表現と $d \bmod (p - 1)$ に基づき $(C^d \bmod p) * B \bmod p$ 又はこれにpを加算した値の剰余系表現を求め、qについても同様にする（123、124）。得られた両剰余系表現に基づき法 $p * q$ の元で $C^d$ と合同である $m'$ の剰余系表現を求める（123～126）。 $m'$ の剰余系表現を2進数表現に変換する（128）。得られた $p * q$ 未満の $m'$ の値又は $p * q$ 以上の $m'$ から所定回数 $p * q$ を減じることによって得た $p * q$ 未満の値を平文mとして出力する（130）。

【選択図】 図3

出 願 人 履 歴 情 報

識別番号 [000003078]

1. 変更年月日 1990年 8月22日  
[変更理由] 新規登録  
住 所 神奈川県川崎市幸区堀川町72番地  
氏 名 株式会社東芝
2. 変更年月日 2001年 7月 2日  
[変更理由] 住所変更  
住 所 東京都港区芝浦一丁目1番1号  
氏 名 株式会社東芝